

Integrating Coding and Language Arts: A View into Sixth Graders' Multimodal and Multilingual Learning

LAURA ASCENZI-MORENO ■ ASHLEY GÜILAMO ■ SARA VOGEL

What's next in language arts? Two trends shape the field today: the increasing need to create, understand, and mobilize the digital literacies that students navigate in their everyday lives (NCTE, 2018), and the need to build on students' language and cultural experiences to transform educational practices (García & Li Wei, 2014; Lenters, 2019; Paris & Alim, 2017). These two trends dovetail in the Computer Science (CS) for All movement, which has pushed states and localities in the United States to provide students from diverse backgrounds with opportunities not just to learn how to use software (like desktop publishing tools), but to *make* software—to program computers and to apply “computational thinking” practices like abstraction and debugging (Brennan, Balch, & Chung, 2014) to problems and topics across disciplines.

There are many visions guiding the CS for All movement and a variety of ways to implement it in and outside of schools (Vogel, Santo, & Ching, 2017). One key trend in CS for All making its way into language arts classrooms is “creative computing” (Brennan et al., 2014), an approach to computing education that views computer code—the commands in a programming language that make a computer work—as having the same expressive capabilities as a paintbrush or a pencil (Resnick, 2006). Through the creative computing approach in language arts, code can be used to compose digital stories and other projects, like games and animations, that can be integrated with literacy development. When combined with culturally sustaining approaches to education (Paris & Alim, 2017), creative computing ensures that students with different languages, backgrounds, and ways of knowing see themselves as authors with text and code.

In this article, we share examples from our project, Participating in Literacies and Computer Science

(PiLaCS), which focuses on how students' language practices shape their participation and engagement in language arts projects that integrate code. The students we work with are middle level emergent bilinguals, or students who use two or more languages in their daily lives. With their teacher, Ashley Güilamo (a member of this project and a coauthor of this paper), students compared and contrasted three ways of telling a story—

one with their bodies, one with text, one animated with code. They also used code, their voices, and images to create animated personal narratives about their journeys to school in the United States and in the countries from which many of them had immigrated.

We found that as students used code to analyze, tell, and share stories, they also did what researchers García and Li Wei call *translanguaging* (2014)—using their linguistic and

semiotic repertoires flexibly and holistically—to engage in language arts competencies such as learning vocabulary, comparing and contrasting texts, retelling experiences, and composing new texts. When students combined multimodal and multilingual resources in these CS-integrated language arts contexts, they produced what we term *dynamic expressions*. Validating, honoring, and making space for students' dynamic expressions in instruction are ways to broaden who can engage and participate in literacy meaning-making, in our case, to include emergent bilingual students.

Integrating Code into Language Arts: Ashley's Multimodal Translanguaging Approach

Tasked with fulfilling her school's commitment to CS for All within her sixth grade bilingual language arts class, Ashley chose to teach a unit with a software

*Through the creative
computing approach in
language arts, code can
be used to compose digital
stories and other projects,
like games and animations,
that can be integrated with
literacy development.*

and programming language called Scratch, created at the Massachusetts Institute of Technology, to support creative approaches to code (<https://scratch.mit.edu/>). Scratch is a blocks-based programming language (Figure 1), which means its users (children and young adolescents) can worry less about causing bugs with minor syntax errors in their code and focus on creating animations, digital stories, and other original projects. They can use characters and backdrops found on the Scratch platform as well as photographs, drawings, and music from the internet. Members of the Scratch community can share their projects, comment on each other's work, and "remix" others' projects. Ashley chose Scratch as a platform for her students because it is browser-based, freely available, and the interface can be translated into more than fifty languages.

Ashley started planning her CS-integrated language arts unit by reflecting on her students' immigration experiences, identities, and language practices. She asked herself a series of questions, included at the end of this article, regarding their language and literacy experiences. Most of her students had arrived in New York City from the Dominican Republic within the last three years. They used Spanish and English daily in and out of school. Ashley identified not only her students' school-bounded literacies but also resources that stemmed from

home, community, and youth culture (students enjoyed watching Spanish language TV, playing video games, and sports). Ashley's centering of students' experiences in her design emerges from translanguaging pedagogy (García, Johnson, & Seltzer, 2017) and research on multiple modalities (Lenters, 2018) that argue that students learn best when they engage all their language resources and their bodies to make meaning.

Ashley then considered how students could grapple with the idea of code as a medium for creative expression. She and our project are guided by a philosophy from computer science called "literate programming" (Knuth, 1984)—the idea that code is not experienced in a vacuum but is always part of a conversation, for instance in a community of digital artists. Making explicit to students the idea that code can be annotated, discussed, and used in a story is important if we want them to view code as connected to language, literacy, and other aspects of their lives and communities. It is from this pedagogical framework that Ashley and other teachers in our project designed activities to build on what students brought to the table, and to develop their language arts and computing skills. What follows are examples from Ashley's class that demonstrate how a CS-integrated language arts curriculum provided her students with space to engage, create, and communicate using language, text, and their bodies in dynamic expressions.

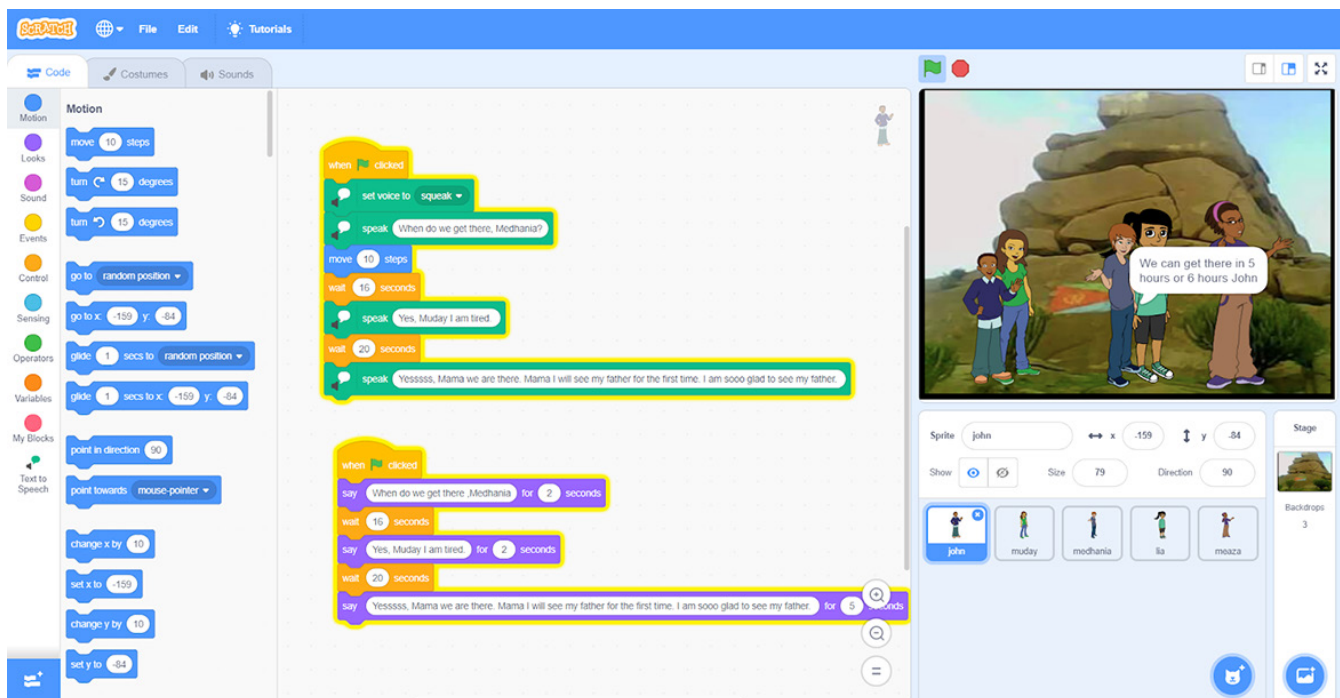


FIGURE 1. Different parts of the Scratch interface. On the left, users can select different code blocks by clicking and dragging them to the center workspace, where users can assemble code to program a project. On the right, students can view their digital stories and animations resulting from their programming.

Comparing and Contrasting Ways to Tell a Story through Language, Movement, and Code

To introduce students to the expressive potential of code, Ashley asked them to compare a genre that would be familiar—the *telenovela* (“soap opera” in Spanish)—to digital stories told in the Scratch programming environment. To warm them up, she asked students to act out a telenovela scene using a script and to watch an animation depicting the same skit in Scratch—all in Spanish. She then asked them to match the script that telenovela actors are given to the commands in code blocks by first color-coding the script according to type of action performed (speaking, moving, or making noise) and then drawing a line from the script to the corresponding code blocks (see Figure 2 for the student worksheet). Some matches were easy—the code block that directed a character to say, “Tú me traicionaste,” (you betrayed me) was an obvious corollary to where the actor says that line in the script. Some were not as straightforward. Álvaro (all student names are

pseudonyms) and his partner were stumped as they tried to figure out how to match a code block that directed a character to *deslizar* (glide) to an action from the script. As Ashley tells it,

Eventually, the boys worked it out and I wanted to know how. Álvaro explained, “Deslizar es como cuando estás jugando béisbol y . . .” (Deslizar is like when you play baseball) and then he acted as if he were playing baseball in front of the room. He swung an imaginary bat to hit the ball and slid to a base.

Through Álvaro’s use of his body and words in his problem-solving, the class gained a strong understanding of the word *deslizar* and how it is used as a code to tell stories in Scratch.

In this anecdote, we see how through actively comparing and contrasting different forms of a story that were written in text, performed, and coded, Álvaro was able to engage in learning new vocabulary and to layer his understanding of the concept of sliding or moving across a screen or stage as something that could be written, performed, or coded. Álvaro’s dynamic

El guión de la telenovela	Los programas en Scratch
<p>Subraya el guión abajo con los siguientes colores:</p> <p>Morado: algo dicho Azul: movimiento Rosado: ruidos/sonidos **si no sabes, usa un lápiz o lapicero normal</p>	
<p>Amigo/a 1: No lo puedo creer. ¡Me traicionaste! ¡Mi mejor amigo/a!</p>	<p>Amiga1:</p> <p>al presionar</p> <p>decir No lo puedo creer. ¡Me traicionaste! ¡Mi mejor amigo/a! por 2 segundos</p> <p>esperar 3 segundos</p> <p>decir No sé quiéres tú. Eres la única persona que sabía la identidad real de mi padre. Y</p> <p>esperar 3 segundos</p> <p>decir Me tengo que ir al Polo Norte por 2 segundos</p> <p>tocar teclado: letra: y esperar</p> <p>deslizar en 1 segs a X: 240 Y: -120</p>
<p>Amigo/a 2: ¡No fui yo! ¡Te lo prometí!</p>	
<p>Amigo/a 1: Yo sé que eras tú. Eres la única persona que sabía la identidad real de mi padre. Y ahora todo el mundo sabe que...</p>	<p>Amiga2:</p> <p>al presionar</p> <p>esperar 3 segundos</p> <p>decir ¡No fui yo! ¡Te lo prometí! por 2 segundos</p> <p>esperar 4 segundos</p> <p>decir Que tu padre es Papá Noel... por 2 segundos</p> <p>esperar 7 segundos</p> <p>cambiar disfraz a every-b</p>
<p>Amigo/a 2: Que tu padre es Papá Noel...</p>	
<p>Amigo/a 1: Me tengo que ir al Polo Norte...</p>	
<p>[Amigo/a 1 empieza a llorar, y corre al otro lado del escenario]</p>	
<p>[Amigo/a 2 gira hacia amiga 1 y cruce los brazos]</p>	

FIGURE 2. This is an image of Álvaro’s completed worksheet for matching text to code. On the left is the script for the skit in Spanish. On the right is the code that animates the same script. The highlights indicate how Álvaro associated each of the lines of text with actions (speaking, making noise, and moving). The connecting lines show how Álvaro matched the actions in the text to code. The circled code is the *deslizar* (slide) block that Álvaro matched to the stage notes “corre al otro lado del escenario” (run to the other side of the stage).

expression of sliding across the room animated his understanding of the connection between the text and the code, showing how integrating code into language arts provides a forum for students' language practices to be integrated and validated. In the next section, we profile how another student in Ashley's class combined oral language, drawing, and playing with objects to create a personal narrative.

Composing a Personal Narrative through Dynamic Expressions

During Ashley's second CS-integrated storytelling project, *Journeys to School*, her multimodal, translinguaging approach invited students' dynamic expressions to help them create narrative games on Scratch representing their journeys from home to school. The project was collaboratively developed by Ashley and Sara (a researcher and coauthor of this article). The literacy goals for this project were centered on students creating a personal narrative and communicating these narratives through a variety of methods—visual prototypes with objects, discussion, and code.

Ashley first asked students to observe and discuss photographs of students from other countries going to school. Students then reflected on their own lives and journeys to school, both in other countries and the United States. Students worked to prototype their games with paper and small objects first, identifying paths, landmarks, and obstacles. After sharing their prototypes with classmates and receiving feedback, students made necessary changes, which resulted in clearer visions for the digital versions of their projects. Throughout the process, students engaged in storytelling while using their own language and social practices—they could speak using language of their choice and could connect their lives to the project while exploring code and literacy concepts.

We focus on Mariposa, who was born in the Dominican Republic and had been in the United States

The integration of code into language arts in ways that promote students' dynamic expressions and meaning-making is a new frontier.

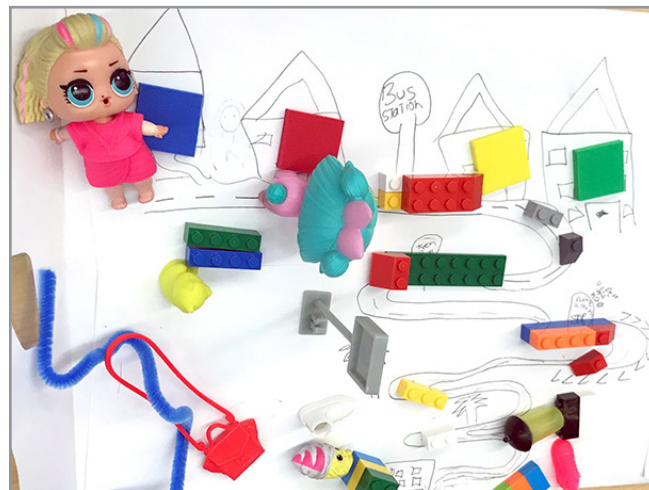


FIGURE 3. *Mariposa's play-based prototype for the Journey to School project.*

for about two years at the time of the project. Mariposa's prototype included pencil drawing and objects from school, as well as toys she brought from home for this activity. As she figured out the challenges that her character would encounter on her journey to school, Mariposa moved the figures and placed objects along the path (Figure 3). When it was time for Mariposa to explain her work to her partner, which she did in Spanish and through movements, her partner commented on the decorative aspect of her prototype. Her partner asked

Mariposa, in Spanish, "Why have so many stores along the journey to school route if they are not necessary to the game?" Without losing a beat, Mariposa noted that her family owns a deli where she sometimes goes in the morning before school starts, demonstrating that while decorative, the objects also reflected her reality and

a potential component of her game to be programmed later. A glimpse into how Mariposa approached this project provides insight into how her language practices rest upon her calling forth her prior experiences, people, and objects to make sense of and create through genres (personal narrative) and media (a digital game).

CONNECTIONS FROM readwritethink

This Strategy Guide series provides guides on teaching with technology.

Lisa Storm Fink

www.ReadWriteThink.org

<http://bit.ly/1sRxAVh>

The integration of code into language arts in ways that promote students' dynamic expressions and meaning-making is a new frontier. Making room for a variety of literacies as well as broadening who and how students can participate in language arts conversations are vital ways to integrating technology equitably into literacy instruction (NCTE, 2018).

Code and Language Arts in Your Classroom

How might similar learning experiences spread across language arts classrooms? Ashley started designing a CS-integrated language arts curriculum by reflecting on her students. Practitioners can similarly ask themselves: Where are my students from? How long have they been in the United States? What languages do they speak? What videos do they watch? What music do they like? What is the nature of their interaction with digital learning? These questions can be starting points to designing CS-integrated language arts curriculum that provides students opportunities for multilingual and multimodal meaning-making.

Ashley also created spaces for students' *dynamic expressions* (ways of making meaning with text, paper prototypes, oral language, and code) and aimed to learn from these. Practitioners can reflect upon how students' dynamic expressions unfold and impact their classrooms with questions like: What am I learning from students' creative meaning-making as they code in language arts? How does my understanding of students' resources and language practices help me make space within my language arts instruction for students' dynamic expressions?

Through Álvaro and Mariposa's stories, we see that integrating code into language arts curriculum in a way that honors students' multilingual and multimodal dynamic expressions enables them to participate in and create new ways of engaging in literacy (Vogel et al., 2019). This type of integrated learning demonstrates how language arts is not about learning discrete skills and competencies, but instead about how language is situated in the world and how it evolves and mixes with various literacies—including CS and code—different modalities, and language practices. Bringing these kinds of integrated experiences to the classroom is the next frontier for language arts instruction. Teachers now have the opportunity to envision how, through

this integration, the language arts space can welcome students' dynamic expressions as a powerful way to connect to, analyze, and engage in a variety of text-based and digital literacies.

ACKNOWLEDGMENTS:

Thank you to our coresearchers on the PiLaCS team—Chris Hoadley, Jasmine Ma, Sarah Radke, Marcos Ynoa, and Kate Menken. We are also grateful for the school

administrators, other teachers, and students at our partner schools who make this work possible and for the anonymous reviewers for their helpful comments.

This research is sponsored by the National Science Foundation under NSF grants CNS-1738645 and DRL-1837446. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Teachers now have the opportunity to envision how . . . the language arts space can welcome students' dynamic expressions as a powerful way to connect to, analyze, and engage in a variety of text-based and digital literacies.

REFERENCES

- Brennan, K., Balch, C., & Chung, M. (2014). Creative computing: Scratch curriculum guide. scratched.gse.harvard.edu/guide/
- García, O., Johnson, S., & Seltzer, K. (2017). *The translanguaging classroom: Leveraging student bilingualism for learning*. Caslon Publishing.
- García, O., & Wei, L. (2014). *Translanguaging: Language, bilingualism and education*. Palgrave Macmillan Pivot.
- Knuth, D. E. (1984). Literate programming. *The Computer Journal*, 27(2), 97–111. <https://doi.org/10.1093/comjnl/27.2.97>
- Lenters, K. (2018). Multimodal becoming: Literacy in and beyond the classroom. *The Reading Teacher*, 71(6), 643–649.
- Lenters, K. (2019). Nerdisms, Almina, and the petsitter: Becoming social commentary composers. *Research in the Teaching of English*, 53(4), 363–389.
- National Council of Teachers of English. (2018, October 25). *Beliefs for integrating technology into the English language arts classroom*. <https://www2.ncte.org/statement/beliefs-technology-preparation-english-teachers/>
- Paris, D., & Alim, H. S. (2017). *Culturally sustaining pedagogies: Teaching and learning for justice in a changing world*. Teachers College Press.

Resnick, M. (2006). Computer as paint brush: Technology, play, and the creative society. In D. G. Singer, R. M. Golinkoff, & K. Hirsh-Pasek (Eds.), *Play = learning: How play motivates and enhances children's cognitive and social-emotional growth*, (pp. 192–208). Oxford University Press.

Vogel, S., Hoadley, C., Ascenzi-Moreno, L., & Menken, K. (2019). The Role of Translanguaging in Computational Literacies: Documenting Middle School Bilinguals' Practices in Computer Science Integrated Units. *Proceedings of the 50th ACM*

Technical Symposium on Computer Science Education, 1164–1170. <https://doi.org/10.1145/3287324.3287368>

Vogel, S., Santo, R., & Ching, D. (2017, March). Visions of computer science education: Unpacking arguments for and projected impacts of cs4all initiatives. In *SIGCSE '17: Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, (pp. 609–614). <https://doi.org/10.1145/3017680.3017755>

Laura Ascenzi-Moreno is an associate professor and bilingual program coordinator at Brooklyn College, City University of New York.

Ashley Güilamo is a sixth-grade bilingual teacher in a New York City public school.

Sara Vogel holds a PhD in urban education and is a postdoctoral fellow at New York University.

ACKNOWLEDGMENTS:

Thank you to our coresearchers on the PiLaCS team—Chris Hoadley, Jasmine Ma, Sarah Radke, Marcos Ynoa, and Kate Menken. We are also grateful for the school administrators, other teachers, and students at our partner schools who make this work possible and for the anonymous reviewers for their helpful comments.

This research is sponsored by the National Science Foundation under NSF grants CNS-1738645 and DRL-1837446. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Reproduced with permission of copyright owner. Further reproduction prohibited
without permission.